UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICANT(S)   William C. Moyer   GROUP ART UNIT: 2111

APPLN. NO.:   10/657,593   EXAMINER: Kenneth S. Kim

FILED: September 8, 2003

TITLE: DATA PROCESSING SYSTEM HAVING INSTRUCTION

               SPECIFIERS FOR SIMD OPERATIONS AND METHOD THEREOF

AMENDMENT

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Responsive to the Office Action dated February 15, 2006, and Examiner's comments with regard thereto, please enter the following amendments in the above-entitled application, without prejudice or disclaimer.

IN THE CLAIMS:

1. (currently amended) A data processing system comprising:

a memory for storing operands;

at least one general purpose register; and

processor circuitry for executing at least a first instruction and a second instruction

subsequent to the first instruction, the first instruction ~~transferring~~ initiating

transfer of a stream of data elements ~~between~~ from the memory ~~and~~ to be queued

in the at least one general purpose register ~~and to be queued in at least one of the~~

~~memory and the at least one general purpose register~~, wherein the second

instruction comprises at least a first source operand, and conditionally ~~dequeueing~~

loading a portion of the stream of data elements from the memory into the at least

one general purpose register based on the at least one general purpose register

appearing as a source operand of the second instruction.

2. (currently amended) The data processing system of claim 1 wherein conditionally ~~dequeueing~~
loading the portion of the stream of data elements is performed when the at least one general
purpose register is used as a source operand for a predetermined type of function specified by the
second instruction.

3. (currently amended) The data processing system of claim 1 wherein conditionally ~~dequeueing~~
loading the portion of the stream of data elements is performed based on a value of a control
field of the second instruction.

4. (currently amended) The data processing system of claim 1 wherein the second instruction
further comprises a second source operand, and the conditional ~~dequeueing~~ loading is performed
when the at least one general purpose register appears as the first source operand, and the
conditional ~~dequeueing~~ loading is not performed when the at least one general purpose register
appears as the second source operand.

5. (currently amended) A method ~~of selectively dequeueing data elements~~ in data processing system comprising:

      providing a memory for storing operands;

      providing at least one general purpose register; and

      executing at least a first instruction ~~and a second instruction subsequent to the first instruction~~;

      initiating transfer of ~~transferring~~ a stream of data elements ~~between~~ from the memory ~~and~~ to be queued in the at least one general purpose register in response to the first instruction;

      ~~queueing the stream of data elements in at least one of the memory and the at least one general purpose register~~;

      executing a second instruction subsequent to the first instruction, the second instruction comprising at least a first source operand; and

      conditionally ~~dequeueing~~ loading a portion of the stream of data elements from the memory into the at least one general purpose register based on the at least one general purpose register appearing as the source operand of the second instruction.

6. (currently amended) The method of claim 5 further comprising conditionally ~~dequeueing~~ loading the portion of the stream of data elements when the at least one general purpose register is used as a source operand for a predetermined type of function specified by the second instruction.

7. (currently amended) The method of claim 5 further comprising

      providing a second source operand within the second instruction; and

      conditionally ~~dequeueing~~ loading the portion of the stream of data elements when the at least one general purpose register appears as the first source operand, and not performing conditional ~~dequeueing~~ loading when the at least one general purpose register appears as the second source operand.

8. (currently amended) A data processing system comprising:

a memory for storing operands;

at least one general purpose register; and

processor circuitry for executing a plurality of instructions, a first one of the plurality of instructions ~~transferring~~ initiating transfer of a stream of data elements ~~between~~ from the memory ~~and~~ to be queued in the at least one general purpose register ~~and to be queued in at least one of the memory and the at least one general purpose register~~, and conditionally ~~dequeueing~~ loading a portion of the stream of data elements from the memory into the at least one general purpose register in response to a second one of the plurality of instructions corresponding to a predetermined instruction within a proper subset of the plurality of instructions.

9. (currently amended) The data processing system of claim 8 wherein the processor circuitry further conditionally ~~dequeues~~ loads the portion of the stream of data elements based on the at least one general purpose register appearing as a source operand of the second one of the plurality of instructions.

10. (currently amended) The data processing system of claim 8 wherein the processor circuitry further conditionally ~~dequeues~~ loads the portion of the stream of data elements based on a value of a control field of the second one of the plurality of instructions.

11. (currently amended) A method ~~of selectively dequeueing data elements~~ in a data processing system comprising:

    providing a memory for storing operands;

    providing at least one general purpose register; and

    executing a plurality of instructions, a first one of the plurality of instructions <u>initiating</u> <u>transfer of</u> ~~transferring~~ a stream of data elements ~~between~~ <u>from</u> the memory ~~and~~ <u>to be queued in</u> the at least one general purpose register; <u>and</u>

    ~~queueing the stream of data elements in at least one of the memory and the at least one general purpose register; and~~

    conditionally ~~dequeueing~~ <u>loading</u> a portion of the stream of data elements in response to a second one of the plurality of instructions corresponding to a predetermined instruction within a proper subset of the plurality of instructions.

12. (currently amended) The method of claim 11 further comprising further conditionally ~~dequeueing~~ <u>loading</u> the portion of the stream of data elements based on the at least one general purpose register appearing as a source operand of the second one of the plurality of instructions.

13. (currently amended) A data processing system comprising:

    a memory for storing operands;

    at least one general purpose register; and

    processor circuitry for executing a plurality of instructions, a first one of the plurality of instructions ~~transferring~~ <u>initiating a transfer of</u> a stream of data elements ~~between the memory and~~ <u>from</u> the at least one general purpose register ~~and~~ <u>to be queued in</u> ~~at least one of~~ the memory ~~and the at least one general purpose register~~, conditionally ~~enqueueing~~ <u>storing</u> a portion of the stream of data elements <u>to the memory</u> ~~based on the at least one general purpose register appearing as a destination operand of a second one of the plurality of instructions~~ <u>in response to the second one of the plurality of instructions corresponding to a predetermined instruction within a proper subset of the plurality of instructions.</u>

14. (currently amended) The data processing system of claim 13 wherein ~~enqueueing~~ conditionally storing is performed based on the at least one general purpose register appearing as a destination operand of the second one of the plurality of instructions ~~in response to the second one of the plurality of instructions corresponding to a predetermined instruction within a proper subset of the plurality of instructions~~.

15. (currently amended) A method ~~of selectively enqueueing data elements~~ in a data processing system comprising:

        providing a memory for storing operands;

        providing at least one general purpose register; and

        executing a plurality of instructions, a first one of the plurality of instructions ~~transferring~~ initiating transfer of a stream of data elements ~~between the memory and~~ from the at least one general purpose register ~~and~~ to be queued in ~~at least one of~~ the memory ~~and the at least one general purpose register~~; and

        conditionally ~~enqueueing~~ storing a portion of the stream of data elements to the memory based on the at least one general purpose register appearing as a destination operand of a second one of the plurality of instructions.

16. (currently amended) The method of claim 15 wherein ~~enqueueing~~ storing is performed additionally in response to the second one of the plurality of instructions corresponding to a predetermined instruction within a proper subset of the plurality of instructions.

17. (currently amended) A data processing system comprising:

        a memory for storing operands;

        at least one general purpose register; and

        processor circuitry for executing at least a first instruction and a second instruction subsequent to the first instruction, the first instruction ~~transferring~~ initiating transfer of a stream of data elements ~~between the memory and~~ from the at least one general purpose register ~~and~~ to be queued in ~~at least one of~~ the memory ~~and the at least one general purpose register~~, wherein the second instruction comprises at least a first destination operand, and conditionally ~~enqueueing~~ storing a portion

of the stream of data elements <u>to the memory</u> based on at least one general purpose register appearing as a destination operand of the second instruction.

18. (currently amended) The data processing system of claim 17 wherein the conditional ~~enqueueing~~ <u>storing</u> is performed when the at least one general purpose register is used as the first destination operand for a predetermined type of function specified by the second instruction.

19. (currently amended) The data processing system of claim 17 wherein the second instruction further comprises a second destination operand and the conditional ~~enqueueing~~ <u>storing</u> is performed when the general purpose register appears as the first destination operand and the conditional ~~enqueueing~~ <u>storing</u> is not performed when the general purpose register appears as the second destination operand.

20. (currently amended) A data processing system comprising:

    a memory for storing operands;

    at least one general purpose register; and

    processor circuitry for executing at least a first instruction and a second instruction subsequent to the first instruction, the first instruction ~~transferring~~ <u>initiating a transfer of</u> a stream of data elements ~~between~~ <u>from</u> the memory ~~and~~ <u>to be queued in</u> the at least one general purpose register ~~and to be queued in at least one of the memory and the at least one general purpose register~~, wherein the first instruction further ~~specifying~~ <u>specifies</u> a number of data elements to be transferred, and conditionally ~~dequeueing~~ <u>loading</u> a ~~plurality of data elements from the~~ portion of the stream of data elements based on the at least one general purpose register appearing as a source operand of the second instruction.

21. (original) The data processing system of claim 20 wherein the stream of data elements is further specified by a control field within the first instruction.

22. (original) The data processing system of claim 20 wherein size of the stream of data elements is further specified by the instruction.

23. (original)   The data processing system of claim 22 wherein the first instruction specifies size of the stream of data elements as a field in the instruction.

24. (original)   The data processing system of claim 22 wherein the first instruction specifies size of the stream of data elements by defining a storage location that contains the size of the stream of data elements.

25. (currently amended) A data processing system comprising:

    a memory for storing operands;

    at least one general purpose register; and

    processor circuitry for executing at least a first instruction and a second instruction subsequent to the first instruction, the first instruction ~~transferring~~ <u>initiating</u> <u>transfer of</u> a stream of data elements ~~between the memory and~~ <u>from</u> the at least one general purpose register ~~and~~ to be queued in ~~at least one of~~ the memory ~~and the at least one general purpose register~~, wherein the first instruction further ~~specifying~~ <u>specifies</u> the number of data elements to be transferred, and conditionally ~~enqueueing~~ <u>storing</u> a ~~plurality of data elements from the~~ portion of the stream of data elements <u>to the memory</u> based on the at least one general purpose register appearing as a destination operand of the second instruction.

26. (original)   The data processing system of claim 25 wherein the stream of data elements is further specified by a control field within the first data processing instruction.

27. (original)   The data processing system of claim 25 wherein a size of the plurality of data elements is further specified by the first data processing instruction.

28. (original)   The data processing system of claim 27 wherein the first data processing instruction specifies size of the plurality of data elements as a field in the instruction.

29. (original) The data processing system of claim 27 wherein the first data processing instruction specifies size by defining a storage location that contains the size of the plurality of data elements.

30. (currently amended) A data processing system comprising:

a memory for storing operands;

at least one general purpose register; and

processor circuitry for executing a plurality of instructions, a first one of the plurality of instructions ~~transferring~~ initiating transfer of a stream of data elements between the memory and the at least one general purpose register and to be queued in at least one of the memory and the at least one general purpose register, and conditionally performing at least one of ~~enqueueing~~ storing and ~~dequeueing~~ loading of a portion of the stream of data elements in response to a control field within a second one of the plurality of instructions.

31. (currently amended) The data processing system of claim 30 wherein the processor circuitry further conditionally performs at least one of ~~enqueueing~~ storing and ~~dequeueing~~ loading of the portion of the stream of data elements based on the at least one general purpose register appearing as an operand of the second one of the plurality of instructions.

32. (currently amended) The data processing system of claim 30 wherein the processor circuitry further conditionally performing at least one of ~~enqueueing~~ storing and ~~dequeueing~~ loading the portion of the stream of data elements based on a value of a control field of the second one of the plurality of instructions.

33. (New) The method of claim 5, wherein initiating transfer of the stream of data elements comprises loading an initial portion of the stream of data elements from the memory into the at least one general purpose register.

9

34. (New) The method of claim 11, wherein initiating transfer of the stream of data elements comprises loading an initial portion of the stream of data elements from the memory into the at least one general purpose register.

35. (New) The method of claim 15, wherein initiating transfer of the stream of data elements comprises storing an initial portion of the stream of data elements to the memory.

36. (New) The data processing system of claim 30, wherein initiating transfer of the stream of data elements comprises performing at least one of storing and loading of an initial portion of the stream of data elements.

REMARKS

Claims 1-36 will be remain pending in the current Application upon entering this Amendment. Claims 1-20, 25, and 30-32 have been amended; and claims 33-36 have been added. Applicants submit that the amendments do not add new matter to the current Application. All the amendments herein have been made in order to clarify the claims and not for prior art reasons. Applicants also submit that (1) no amendment made was related to the statutory requirements of patentability unless expressly stated herein, and (2) no amendment made was for the purpose of narrowing the scope of any claim, unless Applicants have argued herein that such amendment was made to distinguish over a particular reference or combination of references.

Rejection of claims 1-32 under 35 U.S.C. 112

Applicant respectfully submits that claims 1-32 are patentable under 35 U.S.C. 112. Applicant submits that the claims clearly point out and distinctly claim the subject matter. Furthermore, the subject matter claimed is clearly described in the specification, such as, for example, on pages 52-56, which clearly describe the dequeueing and enqueueing, as claimed. However, in order to further prosecution and not for patentability reasons under 35 U.S.C. 112, Applicant has amended each of the independent claims (1, 5, 8, 11, 13, 15, 17, 20, 25, and 30) to clarify whether the transfer is from a register to memory or vice versa, and to replace references to dequeueing and enqueueing with loading and storing, respectively. For example, in reference to claim 1, note that the first instruction initiates transfer of a stream of data elements from the memory to be queued in the at least one general purpose register. An example of this first instruction is discussed with respect to the first lstrmvex instruction in the pseudocode of page 52 which his discussed on pages 53-56. Claim 1 also includes conditionally loading a portion of the stream of data elements form the memory into the at least one general purpose register based on the at least one general purpose register appearing as a source operand of the second instruction. An example of this second instruction is discussed with respect to the first vmac instruction in which R1 is used as source1. That is, when R1 is used as source1 of a subsequent instruction, a portion of the stream of data elements is dequeued (loaded into R1). (See, e.g., page 53, lines 12-15 and page 54, lines 3-21.) Similar clarifying amendments were made to claims 5, 8, 11, 13,

11

15, 17, 20, 25, and 30, where some of these claims address dequeueing (i.e. loading) and other address enqueueing (i.e. storing). Amendments to dependent claims were made to maintain consistency with the terminology changes made in the independent claims. Therefore, for at least these reasons, Applicant submits that claims 1-32 (and added claims 33-36) are all patentable under 35 U.S.C. 112.

Rejection of claims 1-32 under 35 U.S.C. 102

Applicant respectfully submits that claims 1-32 are patentable over US Patent No. 4, 677, 547 (hereinafter referred to as Omoda).

For example, with respect to claim 1, claim 1 claims a first instruction initiating transfer of a stream of data elements from the memory to be queued in the at least one general purpose register and a second instruction subsequent to the first instruction where the processor circuitry conditionally loads a portion of the stream of data elements from the memory into the at least one general purpose register based on the at least one general purpose register appearing as a source operand of the second instruction. Applicant submits that Omoda at least does not teach or suggest these elements. The cited sections of Omoda (e.g., col. 1, lines 17, 20, and 23) simply refer to a vector load instruction which loads vectors X and Y into vector registers VR0 and VR1 (or a vector store instruction which stores vector register VR2 to memory.) However, there is no teaching or suggestion of the first and second instructions as claimed. For example, there is not a second instruction which results in a load of a portion of a data stream (initiated by a first instruction) depending on the general purpose register appearing as a source operand in the second instruction. Therefore, for at least these reasons, Applicant submits that claim 1 is allowable over Omoda since Omoda fails to teach or suggest each and every element of claim 1.

Claims 5, 8, 11, 13, 15, 17, 20, 25, and 30 similarly claim different aspects of a first and second instruction, where many of the arguments presented above for claim 1 also apply to these claims. That is, Omoda fails to teach each and every element of each of claims 5, 8, 11, 13, 15, 17, 20, 25, and 30.

Claims 2-4, 6, 7, 9, 10, 12, 14, 16, 18, 19, 21-24, 26-29, and 31-36 each depend directly or indirectly from allowable claim 1, 5, 8, 11, 13, 15, 17, 20, 25, or 30, and are therefore also

allowable for at least those reasons provided above with respect to claims 1, 5, 8, 11, 13, 15, 17, 20, 25, and 30.

Also, note that added claims 33-36 depend off of claims 5, 11, 15, and 30, respectively, and further detail the initiation of the transfer of stream of elements (see, e.g., pages 52-56 of the current specification).

Conclusion

The Office Action contains numerous statements characterizing the claims, the Specification, and the prior art. Regardless of whether such statements are addressed by Applicant, Applicant refuses to subscribe to any of these statements, unless expressly indicated by Applicant.

Applicant respectfully solicits allowance of the pending claims. Contact me if there are any issues regarding this communication or the current Application.

If Applicant has overlooked any additional fees, or if any overpayment has been made, the Commissioner is hereby authorized to credit or debit Deposit Account 503079, Freescale Semiconductor, Inc.

Respectfully submitted,

SEND CORRESPONDENCE TO:

By: _____

Freescale Semiconductor, Inc.
Law Department

Joanna G. Chiu
Attorney of Record
Reg. No.:   43,629

Customer Number: 23125

Telephone: (512) 996-6839
Fax No.:   (512) 996-6854

13